END
DATE
FILMED
82
DTIC

AD A107743

(15)

AIRMICS

US ARMY INSTITUTE FOR RESEARCH
IN MANAGEMENT INFORMATION AND COMPUTER SCIENCE

DTIC
ELECTE
NOV 25 1981

S D

D

ANALYSIS OF THE APPLICATION OF PSL/PSA
FOR MANAGEMENT INFORMATION SYSTEMS
IN THE USACSC ENVIRONMENT

FINAL REPORT - Volume I   AIRMICS   80-8-1

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Analysis of the Application of PSL/PSA For Management Information Systems in the USACSC Environment Volume I		5. TYPE OF REPORT & PERIOD COVERED FINAL
		6. PERFORMING ORG. REPORT NUMBER AIRMICS 80-8-1
7. AUTHOR(s) Martin F. McDonough		8. CONTRACT OR GRANT NUMBER(s) DAAG 29-76-D-0100 Delivery Order 1108 TCN 79-018
9. PERFORMING ORGANIZATION NAME AND ADDRESS LOGICON 18 Hartwell Avenue Lexington, MA 02173		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS PE (62725A) PROJECT NUMBER DY10
11. CONTROLLING OFFICE NAME AND ADDRESS U.S. Army Institute For Research In Management Information and Computer Science O'Keefe Bldg, GIT, Atlanta, Ga 30332		12. REPORT DATE July 80
		13. NUMBER OF PAGES 28
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Approved for public release; distribution unlimited

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

SYSTEM, DATA PROCESSING, REQUIREMENTS, METHODOLOGY PSL/PSA, REQUIREMENTS ENGINEERING

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)
Based on a general awareness of the effects of inadequacies in requirements definition throughout the life cycle of a system, AIRMICS has initiated a comprehensive program for application of the latest methodologies in requirements analysis to the Army's information system development programs. This report is a review of the implementation problems involved in using a current available automated tool and to identify further research topics which will be used to guide continuing research.

DD FORM 1473 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

ANALYSIS OF THE APPLICATION OF PSL/PSA FOR MANAGEMENT

INFORMATION SYSTEMS IN THE USACSC ENVIRONMENT

Final Report - Volume I   Logicon Report No. ESD-R0022

July 1980

For The

U.S. ARMY COMPUTER SYSTEMS COMMAND

Prepared by

18 Hartwell Avenue
Lexington, MA 02173
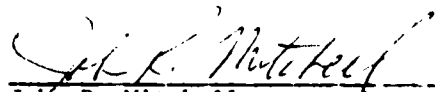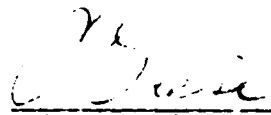
DTIC
ELECTE
NOV 25 1981
S   D
D

This Technical Report has been reviewed and approved.

Daniel E. Hocking
Computer Scientist, Computer
Science Division, U.S. Army
Institute for Research in Management
Information and Computer Science

John R. Mitchell
Chief, Computer Science
Division, U.S. Army
Institute for Research
in Management Information
and Computer Science

Clarence Giese, Director
U.S. Army Institute for Research
in Management Information
and Computer Science

ANALYSIS OF THE APPLICATION OF PSL/PSA FOR MANAGEMENT INFORMATION SYSTEMS IN THE USACSC ENVIRONMENT

Final Report - Volume I   Logicon Report No. ESD-R0022

July 1980

Author: Reviewed by:

Martin F. McDonough   James W. Craft
                                                            Larry A. Johnson
Martin F M'Donough   Daniel G. Smith

The views, opinions and recommendations expressed
in Volumes I and II of this document are those of
the author and do not constitute an official Depart-
ment of the Army position or policy unless so
designated.

LOGICON

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Study Objectives

The objectives of this study were to provide an analysis of the Problem Statement Language/Problem Statement Analyzer (PSL/PSA) requirements analysis tool produced by the University of Michigan's ISDOS project, to determine its worth as a base for further research, and to determine how to best integrate PSL/PSA into the U.S. Army Computer Systems Command (USACSC) environment.

1.2 Project Background

Through its participation in the University of Michigan's ISDOS project, the Army Institute for Research in Management Information and Computer Science (AIRMICS) became interested in a computer-aided technique for structured documentation and analysis of information processing systems (PSL/PSA). Logicon, Inc. has been doing both research and actual systems application with the U.S. Air Force version of the same tool for over 5 years. The Air Force version, the Users Requirements Language/Users Requirements Analyzer (URL/URA) was also developed by the University of Michigan's ISDOS project and is virtually the same as PSL/PSA.

Logicon proposed to assist AIRMICS in the analysis of PSL/PSA as a possible tool to support an ongoing requirements formulation project. The USACSC Vertical Force Development Management Information System (VFDMIS) program was selected to provide a test application of PSL/PSA to the definition and validation of an actual target systems requirements. As part of Logicon's activities, training in the use of PSL/PSA was to be provided to the VFDMIS program personnel.

1.3 Executive Summary

The specific tasks of this study were to provide AIRMICS with an analysis of PSL/PSA and to do the following: 1) to train USACSC personnel in the use of PSL/PSA, 2) to prepare a report evaluating the tool's applicability

1

to the management information environment, 3) to outline ideas establishing controls and procedures for using PSL/PSA, 4) to provide suggestions for improving the tool, and 5) to determine how to use the outputs of PSL/PSA to improve communication within the USACSC environment. Section 3.1 of this report reviews the formal training sessions provided to USACSC by Logicon and the informal consultation on the actual application of the tool. Section 3.2 and the Requirements Engineering Guidebook (Volume II) provide the requirements engineering methodology necessary for the successful application of PSL/PSA in the management information systems environment and for establishing the necessary controls and procedures for the regular use of any such tool by USACSC. This section goes on to evaluate how PSL/PSA can be used to support the concepts of the "System Sketch" as outlined in the paper Requirements Formulation for MIS Through System Sketching by Gabriele, Jazayeri, and Mitchell. It first reviews the current practice of USACSC system developers who are using PSL/PSA then it goes on to discuss how the use of the tool can be improved and finally outlines how the tool could support each of the five techniques of system sketching proposed in the above referenced paper. Section 3.3 is dedicated to applications planning necessary for the successful implementation of the requirements methodology covered in section 3.2 and the guidebook. Section 3.4 outlines ways to improve the utility of PSL/PSA to USACSC. There are three general areas where efforts to improve the utility of PSL/PSA should be concentrated. They are training, user interface and report generation. To improve the training it should be tailored more to the specific program on which the tool is to be employed and provided in the form of both classroom sessions, for the introduction to the tool, and on-the-job-consultation to resolve problems as they arise. The user interface area deals both with the actual use of the tool and the allocation of resources to support the tool. A preprocessor is discussed which would improve the novice users ability to take full advantage of PSL/PSA as well as reduce the burden of training new users. The merits of batch verses interactive use of PSL/PSA are also covered with the pros and cons of each being discussed. The last part of section 3.4 covers extending the report generating capabilities of PSL/PSA to make the tool better able to serve the system developer in the production of large software systems. The

final section, 3.5, describes how the products of PSL/PSA can be used by the developer to keep the proponent and user organization as well as the developers own management informed on the status of progress being made and the quality of the requirements being maintained by PSL/PSA.

2. TECHNICAL APPROACH

2.1 Overview of Technical Tasks

This study was organized around five basic tasks which can be divide into three categories, task-1 training, task-2 evaluation, and tasks-3-4-5 recommendations.

Task-1 required training in both the application of PSL/PSA to the VFDMIS program and in the integration of the automated requirements engineering methodologies of PSL/PSA, with the manual techniques of Structured Analysis and Design of Yourden, Inc.

Task-2 required the evaluation of both PSL/PSA as a tool to support the management information systems environment and of a paper written on the concept of breadboarding or system sketching and to determine how PSL/PSA supports the concepts outlined in the paper.

Tasks-3-4-5 required outlining recommendations for a program to establish controls and procedures necessary for implementing the regular use of an automated requirements analysis tool or tools at USACSC. It required an outline of improvements needed to correct any deficiencies identified during the evaluation of PSL/PSA to improve communications between the developer, the proponent, and the ultimate user of the target system.

2.2 Tasks Inputs

A complete list of all documents used to support this study can be found in Volume II, Appendix A.

2.3 Detailed Task Description

2.3.1 Provide on-site training for USACSC personnel in the use of PSL/PSA - Task 1.

Task 1 concentrated on the preparation and presentation of two formal training sessions and several follow-on consultation sessions. The training itself concentrated on the general concepts of requirements engineering using an automated tool, and on the application of this training to management information systems. The training was also required to integrate automated requirements engineering and structured Analysis and Design.

2.3.2 Prepare a report analyzing the applicability of the tool to the management information systems environment and determine if it supports the concepts outlined for breadboarding in the attached concept paper - Task 2.

Task 2 concentrated on evaluating PSL/PSA to determine its value and place in the USACSC environment. This evaluation served to identify the tool's strong points and weaknesses. Also this task required the evaluation of a paper on the concepts of breadboarding or modeling a target system in order to validate the users requirements prior to the development of the actual target system. And finally this task required the determination of how and where PSL/PSA could support the concepts of breadboarding.

2.3.3 Outline a program for establishing the necessary controls and pro-cedures required for regular use of any such tool in the USACSC environment to include communications among developer, proponent and user and/or alternatively - Task 3.

Task 3 required the evaluation of the USACSC environment and operating procedures and to determine a set of guidelines to support the regular use of PSL/PSA. These recommendations are covered in the Requirements Engineering Guidebook. Also required were recommendations on how PSL/PSA can be used to improve communication among the developer, the proponent, and the user during both the requirements definition and validation phase and during the actual development.

5

2.3.4 Outline a program for correcting the problems which make the tool inappropriate for use - Task 4.

Task 4 concentrated on outlining recommended solutions to PSL/PSA weaknesses which were identified in task 2 of this study.

2.3.5 Determine how to fit the resultant specifications into the three way communications scheme imposed by Army procedure as viewed from the developers viewpoint - Task 5.

Task 5 concentrated on identifying which of the PSA outputs would best serve the USACSC in communicating their development work to both the proponent and the user. It also makes recommendations on how to use PSA output to monitor progress on the development cycle by USACSC management.

3.   STUDY RESULTS

3.1   VFDMIS Training: Requirements Engineering Using PSL/PSA

3.1.1   Introduction

The USACSC personnel assigned to the VFDMIS project decided to employ the University of Michigan ISDOS project's automated requirements engineering tool, PSL/PSA, to support the definition and validation of the VFDMIS system requirements.   Also selected to support this effort were the techniques of Structured Analysis as presented by Mr. Tom DeMarco of Yourdon Inc. in his book Structured Analysis and System Specifications.   It was felt that the tool and the techniques would complement each other and greatly enhance the definition and development of the VFDMIS project.   PSL/PSA and DeMarco's Structured Analysis, although pursuing similar goals, were developed in isolation from each other.   DeMarco defines a manual analysis method using data-flow-diagrams whereas PSL/PSA is an automated data base management system using short English statements with linking relationships.   The problem of effectively integrating the two systems is not defined by either system developer.   Logicon has been working with both PSL/PSA and the concepts of Structured Analysis for over five years and has developed and used a very successful methodology for integrating the two.   Because of this background Logicon was contracted by AIRMICS to provide the USACSC VFDMIS personnel with formal training and consultation on the methodology of Structured Analysis using PSL/PSA.

Actual training on PSL/PSA as a tool was provided by the University of Michigan's ISDOS personnel. Consultation on Structured Analysis was provided by Yourdon Inc.   The training on the methodology of integrating the two was provided by Logicon in two sessions, the content of which are outlined in the following paragraphs.

3.1.2   First Training Session

The first session began with an introduction to Logicon's methodology using a small and easy to comprehend problem.   The recipe for baking

7

bread was used as an example of a specification with the methodology being applied to defining the functions and flow of the process of making bread.

This simple example showed the concepts and terms being used in Structured Analysis, and related them to PSL/PSA. The relationships to PSL/PSA demonstrated both the language features of PSL and the report generation capabilities of PSA which support Structured Analysis. The first training session used both a 35mm slide presentation, which covered requirements engineering and the entire Bake Bread example, and several hardcopy handouts. The first handout was a complete set of training documentation on the Bake Bread example, including how to initiate and build a PSL data base, instructions and examples on how to generate PSA reports, and example results of the outputs from these report generating commands. The second hardcopy training aid was a complete set of documentation similar to the first set, but using the example of the Fleury Merchandise Market used by Yourdon Inc. to teach the Structured Analysis techniques. This second example integrated the requirements analysis techniques with the Structured Analysis techniques.

3.1.3 Second Training Session

The second training session was scheduled a month later to allow the VFDMIS personnel time to become familiar with the data presented and to determine additional areas where training was required to support their particular application. This second session was less formal than the first and consisted primarily of chalkboard presentations supported by a series of student handouts showing selected reports generated from the PSL data base. Each report was the subject of a chalkboard presentation explaining its derivation and utility in the system development life cycle. This format provided for an open exchange between the students and the instructor.

The second session also reviewed the first session and then proceeded to cover three new areas of requirements engineering: requirements traceability, management of the requirements engineering process, and the application of PSL/PSA to the production of Data-Flow-Diagrams (DFDs). To

demonstrate these principles the Fleury Market example was used again in order to maintain continuity in the multiple training sessions. Two separate data bases were created using the case study overview "The Market at Fleury Les Deux Eglises" provided by Yourdon Inc. The first data base represented the top-level requirements of the market. This was done by using selected DFDs and associated data dictionary items from the Yourdon case study overview, and modeling them using PSL/PSA. This top-level PSL/PSA model of the market was presented to the students as one of the handouts. The second data base was developed from the requirements allocated to the next-lower-level DFD. This second data base modeled the requirements in the market which were allocated to the billing-and-accounting function only, and included associated data dictionary items. It also was provided as a student handout. This training session introduced the student to the concepts of applying PSL/PSA to structured DFDs and developing separate data bases for high-level requirements (originating requirements) and lower level requirements (allocated requirements).

3.1.4 Consultation

Logicon provided consultation to the USACSC in a number of areas relating to the installation and application of PSL/PSA. This consultation was provided both on site at Hq. USACSC, Ft. Belvoir, Va., and by telephone. The major emphasis of the early consultation sessions was placed on the selection and use of particular PSL elements for the initial Army application of PSL/PSA. Limiting the number of PSL elements for the initial application served several purposes: the scope of the training could be concentrated; the amount of new material presented to the USACSC personnel would be comprehensible and manageable; the selected elements would provide a solid foundation for future expansion of PSL use; and the reduced size would allow the University of Michigan to produce a 200k-byte version of PSL/PSA which would allow USACSC to have day-time interactive use of the tool. Also, during these early sessions, Logicon began to explore aproaches which would enable the effective implementation of the DeMarco Structured Analysis techniques in PSL/PSA.

LOGICON

Following the two formal training sessions covered in 3.1.2 and 3.1.3 above, Logicon continued to provide consultation on the application of PSL/PSA to USACSC software development programs. The VFDMIS program represented the only program within USACSC where PSL/PSA was being applied to the requirements definition and validation process. The VFDMIS group was developing actual hands-on experience with PSL/PSA and was encountering areas in their day-to-day application of this new technology requiring consultation. Personnel from the STANFINS program were considering the use of PSL/PSA in the same role as the VFDMIS program. Logicon provided the STANFINS personnel with consultation on what it would require to use PSL/PSA and how to plan for its use.

3.2   Requirements Engineering Methodology

3.2.1   Introduction

The object of any requirements engineering methodology is to produce quality requirements in a structured manner. The term "structure", as it applies to requirements definition, means "the arrangement or interrelation of the system parts in a definite pattern of organization as dominated by the general character of the whole system". Any Management Information System (MIS) can and should be defined in a structured manner.

In the traditional sense of system development life cycle, six prime activities take place: conceptualization, requirements definition and analysis (requirements engineering), design, development, testing and installation. The first activity which lends itself to a structured approach is the requirements engineering/analysis, which, if done properly, will insure the feasibility and completness of the conceptualized system and support all subsequent activities. In order to apply a structured approach to these activities, the system development team should produce a requirements engineering plan (Vol II, Section 4.3) which documents the specific approach to be taken in the development of the target system. The plan should take into account, among other things, all available resources, methodologies and tools to be used, and establish how they are to be integrated into the development life cycle. It is here that the synthesis of the output of each development stage with the input to the next stage is accomplished.

There is a limited but increasing number of structured methodologies available in the market place. They all fall into the general catagories of top-down design, structured programming and/or structured analysis.

> HIPO (IBM)
> Composite Design (Meyers/Constantine)
> Structured Analysis and Design Technique (D. Ross)

System Design and Documentation (H. Katzan)
Structured Analysis and System Specification (T. DeMarco)
Structured Design (Yourdon)
Structured Design (Warnier/Orr)
Jackson Structured Design (M. Jackson)
Structured Analysis and Design (C. Gane)
Structured Systems Analysis (Gane/Sarson)

The requirements engineering methodology for the MIS application discussed in this report (Vol II) does not emphasize one of the above methodologies over the others. Rather, it provides the framework and guidance necessary for the incorporation of one or more of these structured methodologies into the system development process. It also provides the guidance for the integration of these methodologies with automated tools (such as PSL/PSA) which support and record the development of target system requirements.

In the most general terms, systems are a mixture of information or data flow, and functional or control flow. The requirements for either of these areas can be structured both hierarchically and sequentially. It is the general consensus that for an MIS development the primary analysis should be placed on data-structure and data-flow. This approach is considered to be problem oriented and generally leads to alternative design solutions as opposed to imposing preconceived design solutions during the initial phases of development. Data-flow analysis and data structuring, using PSL/PSA, are covered in detail in Volume II Sections 4.9 and 4.10, respectively.

The functional decomposition and control-flow analysis is generally thought of as solution-oriented. They concentrate primarily on the processes to be performed and the sequence and hierarchical relationships between them. Functional decomposition and control-flow analysis, using PSL/PSA, are covered in detail in Volume II Sections 4.5 and 4.11, respectively. All systems consist of both data and control flows. An example of a system in which control flow would take priority over data-flow would be a weapon's fire control system. Here the sequence of events is of the utmost importance and the system would deal with small amounts (if any) of data.

12

3.2.2   Requirements Engineering Methodologies for MIS Applications.

This section summarizes Volume 2 of this report: "Requirements Engineering Guidebook, Requirements Engineering Using an Automated Tool: PSL/PSA". The requirements engineering guidebook provides guidance and standards for defining and analyzing the requirements for a target system using an automated tool (PSL/PSA). The guidebook is a product of over five years of experience in the application of automated tools to the definition and development of major military and government ystems. It addresses the process of functional decomposition and data-structuring of the target systems requirements. The requirements engineering methodology presented in the guidebook is compatable with modern structured approaches to requirements definition and analysis (see 3.2.1). It also provides the necessary guidance for the integration of one or more structured approaches with the automated tool PSL/PSA.

The methodology in this guidebook divides systems definition along two major axes: data-structuring and functional decomposition, with data-structuring being the emphasis in MIS development. Data-structuring analysis is the process of defining the composition and flow of data into, within, and out of the system. During this analysis, the flow relationships between external system inputs and resulting system outputs are identified in data flow diagrams. Methods are provided which permit the analyst to determine relationships between associated functions and the internal data necessary to support the derivation of the output. A suggested set of PSL statements to support data flow analysis is provided with examples of their application. Data-structuring of both external and internal information and the development of a data dictionary are explained, and a set of PSL statements to support this activity is also provided, with examples. The functional decomposition analysis concentrates on developing the hierarchical and sequential relationships of the system's functions. The functional hierarchy provides a view of the system, starting in the analysis phase, as an aggregate of functions broken down into a logical arrangement of subordinate discrete activities which must be performed. The sum of the functions on a

given level of a branch are equal to the function at the next higher level of that branch. This principle means that the total system activities are defined by the functions at the lowest level in the hierarchy (functional primitives). The sequential relationships of the functions are determined through control-flow analysis. This analysis provides a means of viewing the system from an activity-oriented perspective and is often referred to as functional-flow analysis. A suggested set of PSL statements to support control-flow analysis is provided along with examples of their application. The resulting control-flow diagrams describe the order in which functions are to be activated, whereas the data-flow diagrams do not indicate any preferred ordering of functions. It is recommended that the final ordering of functions be prepared after the completion of the data flow analysis. This will prevent preconceived notions of design from being imposed during the analysis phase.

3.2.3 Applicability of PSL/PSA to the System Sketch

The following abstract is from the paper REQUIREMENTS FORMULATION for MIS THROUGH SYSTEM SKETCHING by Gabriel, Jazayeri, and Mitchell:

> Requirements formulation is defined as that stage in the software life cycle when a user recognizes and organizes a need for a software system in his mind and transforms it into a written requirements document. A system sketch is proposed as a tool in helping the user and developer formulated and validate system requirements.

The paper outlines the phases of requirements engineering and goes on to discuss the idea of the "System Sketch". Five candidate techniques for producing a System Sketch are briefly discussed; they are, automatic programming, software tools (reuseable modules), the general system, throw-away coding, and functional simulation.

The prime purpose of the System Sketch is to provide a common means of communications between the target system user and the developer. This communications medium is needed to help formulate and validate the users' requirements. In the paper the authors defined five components necessary for a users' Initial Requirements Statement document. They are: system outputs, output formats, system inputs, input formats and performance criteria. The user can reasonably be expected to have a good knowledge of these five components; the problem is to get the user to communicate them clearly. Knowledge of the processes by which a computer based system can effectively fulfill the users' requirements is in the domain of the system developer. The synthesis of these two areas of expertise is the key to quality systems development. The remainder of this section discusses how PSL/PSA can be employed to support both the communications and synthesis of ideas and the concept of the System Sketch in the formulation and validation of the users' requirements.

15

In current practice, the VFDMIS developers at USACSC are working directly with the users, employing a modern Structured Methodology (see 3.2.1) supported by data flow diagrams to both formulate and validate the users' requirements. This is a slow manual process which consumes a large number of labor hours, and direct interaction with the users may not always be possible. Following this activity, selected portions of the requirements are recorded in a PSL data base for further analysis. This process of requirements definition appears to be simultaneously traversing both axes of the requirements engineering methodology covered in Volume II of this report (see 3.2.2 in this volume). Although the activities on both axes are highly interrelated and performed in tandem, the developer must be careful not to produce preconceived designs ahead of a complete understanding of the data-flow/data-structure requirements.

PSL/PSA can be utilized to a greater extent than it currently is in the USACSC environment and in so doing it will move USACSC closer to the concepts proposed under the System Sketch. The target system's inputs and outputs, lie along the data-flow/data-structuring axis of the requirements engineering methodology defined in Volume II Sections 4.9 and 4.10. The selected PSL statements provided in Sections 4.9 and 4.10 are sufficient to allow the user to document the five components of information he is being asked to provide in the Initial Requirements Statement. The use of PSL will force the user to clearly think out each requirement thus eliminating much of the ambiguity or haziness prior to committing them to paper. The guidebook (Volume II) provides examples and techniques for performing data-flow/data-structuring analysis for requirements formulation and validation, and this should be done starting in the earliest stages of analysis.

The development of a user-friendly pre-processor, using something like a menu-board or a stimulus and fill-in-the-blanks type approach to coach the user, would greatly enhance the novice user's ability to make full use of the PSL/PSA capabilities. The results would be a valid, well thought-out set of requirements statements in the form of a structured PSL/PSA data base. Preceding the development of the pre-processor, standards for what

information should be included in the Initial Requirements Statement would need to be established and incorporated into the pre-processor design.

The processes or functions that will fulfill the users' requirements lie along the functional-decomposition/control-flow axis of the methodology defined in Sections 4.5 and 4.11 of the guidebook. The PSL statements and technical guidance provided in these sections, coupled with the existing knowledge of PSL/PSA within USACSC, will allow the fullest use of PSL/PSA by the developer.

With the users' requirements now documented in a structured language (PSL) and maintainable by a computer, the system developer can begin to produce the functional requirements of the target system. The formulation of these requirements using the PSL structured format will allow their immediate synthesis with the data requirements of the user. This process forces the evolution of the total systems' requirements to be accomplished in a structured manner. The use of the analyzer (PSA) will improve the validation of the integrated system requirements as they are developed, insuring that they are complete, consistent and unambiguous. The use of selected PSA-generated reports will support system walk-through reviews with the user during the development of the target system. Some examples of this would be the Structure Report showing the hierarchical decomposition of the system as an aggregate of functions; Content Report showing the hierarchical decomposition of the data (I/O); Process Chain Report showing the sequence of events and functions which results from each event or function specified as input; or the Extended Picture Report showing the derivation of the output starting from the input.

It may require more than the paper outputs of PSA to effectively demonstrate to the users satisfaction that the proposed system will do what the user wants and this requirement could be satisfied by a System Sketch. PSL/PSA will support the five candidate techniques for System Sketching proposed in the Gabriele, Jazayeri, and Mitchell paper, each to a different degree of sophistication. As for "automatic programming", PSL/PSA can be used to

define and record intermodule design relations, and a Program Design Language (PDL) can be used for the actual module design. The PDL statements can be recorded in the PSL/PSA data base under either "Description" or some other appropriate element. A translator interface to the PSL data base could be capable of extracting the required data and feeding it to a code generator. The results of the code execution, the System Sketch, would then be demonstrated to the user to determine if the stated requirements specified his desired system. Any modification needed to the initial requirements statement can easily be made to the PSL data base and recycled through the code generator in an iterative process until the user is satisfied with the current version of the system sketch. Logicon is currently using the PDL concept to automatically produce Fortran code in the development of some of our in-house software tools. The integration of PSL/PSA and automatic programming to support the concept of a System Sketch is a very interesting idea which deserves further research.

PSL/PSA would support the concept of "software tools" (reusable modules) in a less dynamic manner. It can be used to define the requirements of the target system's individual modules which may be fulfilled by existing software modules (Module library). In a totally separate application, PSL/PSA can be used to define and record the functional capabilities of the Module Library itself. The PSL/PSA-produced module requirements for the target system could then be compared to the PSL/PSA-produced Module Library catalog to aid in the identification and selection of Library modules to fulfill the target system's requirements. The level of detail which can be recorded in the Module Library data base would go a long way toward solving many of the problems associated with the concept of a library. Standards could be established to resolve which modules are basic enough to be included in the library, Boolean constructs in PSL/PSA queries will allow the user of the library to automatically determine exactly what is available to fulfill his specific needs, and the module descriptions can aid in determining the suitability of the module.

The normal hierarchical structuring of the target system in PSL/PSA, along

18

with the control flow and information flow analysis required to produce a quality logical model of the target system, fully supports the concept of the "general system". A high level definition of an MIS produced and maintained in a PSL/PSA data base would possess sufficient common characteristics of a generic MIS as to provide the user with a well defined skeletal structure. A PSL/PSA pre-processor could prompt the user and enable him to particularize the system and refine his desired product. Volume II of this report is an in-depth description of how the target system can be structured using PSL and the report generating capabilities of PSA which support the development of a general system.

The target system structure discussed above for the general system will also support the idea of "throw-away-code". Throw-away-coding of the target system will provide the greatest assurance that the requirements meet the users' needs and desires; however, it is also very expensive to build a system only to throw it away.

A simulation capability for PSL/PSA is currently under study by Logicon. Early indications are that a capability to simulate performance based on input, as defined in the PSL data base, may be feasible. However, the shortcoming to this functional simulation is that it will provide information on performance but is not designed to display information going into or out of the system.

There are good points in each of the five concepts proposed in the paper on the System Sketch. A system combining the best of each of the five concepts is feasible and further research into the System Sketch would be very beneficial.

3.3 Applications Planning

In Section 3.2, Requirements Engineering Methodology was discussed. This section addresses the Requirements Engineering Plan necessary for the successful implementation of that methodology.

Management approaches to planning a project will differ from one organization to another and from one project to another within the same organization. There are many factors which influence and make unique the planning of each new project. It is of prime importance to the success of any project, regardless of its uniqueness or the host organization's policies, that all activities be defined and the necessary roles and responsibilities for managing and completing the defined activities be assigned within the project organization. Without a disciplined approach, certain activities will be over-performed, others will be under-performed, and still others will be delayed or not accomplished at all. Section 4.3 of the guidebook addresses the Requirements Engineering Plan.

Like the target system, the development life cycle of that system is made up of inputs (users' requirements and available resources), processes (activities to be performed), and outputs (products that will result in the target system). Similarily, the development life cycle is subjected to the constraints of a set of performance criteria (cost, time, etc). The Requirements Engineering Plan must include the resolution and documentation of several key issues. The first of these issues is that all of the outputs from each stage of the development process and how they will support the following stages must be clearly defined. This definition will cover what is to be produced, its format, and when and where it is to be produced. Next, the available resources or inputs must be identified. This will include at least; available finances, personnel to be involved, the users' initial requirements statement, and the facility resources of the development center. The final and most difficult issue is the planning of how to turn the available input into the desired output -- on time and within cost. Here, specific policies and procedures concerning the allocation and use of

resources, must be established and adhered to as closely as is practical. The development team must be organized with tasks, controls, and responsibilities assigned at all levels.

A structured approach to the development process requires that the tasks be well defined, bounded, and related by hierarchy, sequence, and data dependency. A well defined work breakdown structure with realistic schedules and milestones must be established. A process for monitoring and reporting progress must be established and a formal system of evaluations and progress reviews should be integrated with the work schedules and milestones. Formal tests should be imposed at selected points in the development cycle, such that the results of every major task must pass established criteria before proceeding to the next task or stage. Deficiencies should be formally reported in order to allow for any schedule or work adjustments which may be required to implement a review and changes to, or reaccomplishement of, both current and previous tasks which may be affected. As the system's development cycle progresses the entire Requirements Engineering Plan should be periodically reviewed with proper adjustment being made where necessary.

The use of tools, techniques, and analysis and design methodologies should be established in the plan under policies and procedures. The continued use of Structured Analysis techniques (Yourdon) or similar concepts by USACSC for the decomposition of the target system's functional hierarchy in order to establish the derivation of the system's outputs through information flow analysis should be encouraged. However, any such tool, technique, or methodology should be used only with a good understanding of how it is going to help in fulfilling the objectives of the project. A full definition of how new tools and techniques are to be used and to what level of detail the analysis is to be carried is very important in maintaining their usefulness and in preventing them from becoming an end unto themselves. The leveeing of additional tasks relating to these tools and techniques without this understanding may in fact consume a great deal of resources while producing very limited results.

Volume II of this report, the Requirements Engineering Guidebook, addresses
the definition, analysis, and documentation of a system's requirements using
PSL/PSA. The guidebook has been prepared to assist the future user of
PSL/PSA in applying the tool productively to the logical modeling of a target
system. The intent of the guidebook is to provide the framework of require-
ments engineering and establish the utility of PSL/PSA within that framework.
The guidance provided is intended to aid the new, as well as the experienced,
PSL/PSA user by providing a structured approach to the logical modeling
process and identifing the current features of PSL/PSA best suited to logical
modeling. The guidance is sufficient to support the regular use of PSL/PSA,
or any similar tool, within the USACSC environment. As the user's pro-
ficiency increases, the appendices of the guidebook provide references for
more detailed examples of the language and report features of PSL/PSA. The
guidebook, together with the applications planning outline discussed in this
section, should be used by USACSC for establishing the necessary controls and
procedures required for the regular efficient use of PSL/PSA at USACSC.

3.4 Way to Improve the Utility of PSL/PSA to USACSC

There are three general areas where efforts to improve the utility of PSL/PSA in the USACSC environment should be concentrated. They are: training, user interface, and PSA report generating capabilities. Suggestions for improvements are outlined below.

With reference to the training material outlined in Section 3.1, the area of applications training needs to be improved. Following the formal training which introduces the future PSL/PSA user to the tool and its potential application, expertise in PSL/PSA should be made readily available to the user on a consulting basis. This would allow the user to solve problems as they arise when a solution is more meaningful than if presented in a classroom when the student/user is unaware of the existence of the problems. This expertise is currently available to USACSC from external sources with a depth of knowledge about PSL/PSA and its application to major systems development. As more projects within USACSC use PSL/PSA, eventually a core of expertise internal to USACSC will be developed.

The user interface with PSL/PSA also needs a good deal of improvement. The need for a user friendly pre-processor was discussed in Section 3.2.3. The development of such a pre-processor would alleviate much of the burden of training the new PSL/PSA user. Like the target system's user, the PSL/PSA user should have to know only what data has to be input, its format, what data he wants output, its format, and the approximate time and resources required by the host system to process the activity. This leads to another area of concern, which is turn-around time. The use of batch processing can adversely affect the usefulness of outputs from PSA. Long delays in the receipt of outputs will reduce the user's reliance on the tool, thus having adverse ramifications on the overall practicality of using such an automated tool. Interactive use of PSL/PSA solves the problem of turn-around time and increases the user's reliance on the tool, but it also increases the demand for higher use of the host system's resources. Current interactive users of PSL/PSA claim they would not care to use the tool in batch mode; however,

users who are forced to use the tool in batch mode due to limited resources feel that it still fulfills their needs and prevents the actual use of the tool from gaining too high of a priority in the user's allocation of time. The actual decision on batch or interactive, if not forced by the availability of resources, should be resolved based on how PSL/PSA is to be employed in the overall development program. If the goal is to have a high use of the tool or a time constraint exists then interactive use is more desirable. If the goal is a low rate of use with more calendar time available, then batch processing may be sufficient to support the program. More typically, a reasonable mix of interactive and batch will result in the most effective use of PSL/PSA.

Several report generating extensions to the DoD version of PSL/PSA have been developed by Logicon to provide the tool with greater utility in the large system development environment. These extensions can be added to the core PSL/PSA package without adversely affecting the tool's performance. These extensions are described and demonstrated in Volume II Appendix D of this report. An example would be the Extended Structure Report (Vol II, Appendix D, Figure 3). This report adds to the basic PSA structure Report the option of displaying all information related to the functional requirement; this would otherwise require the user to coordinate the outputs from multiple PSA reports. This report provides the user, in one hierarchically structured report, a much more informative view of the target system's requirements as defined in the PSL data base. The Data Base Status Report (Vol II, Appendix D, Figure 2) is another example of a Logicon extension. This report prints all of the requirements in the data base in the order in which they appear in the source documentation. It provides the complete status of each requirement and the overall status of the data base. It also provides a cross-reference to where each requirement is to be found in the Structure Report. A third example is the Requirements Traceability Report (Vol II, Appendix D, Figure 9). This report traces each requirement to its source document(s) and traces orginating requirements to the allocated requirements or to the design modules, as they are represented in separate data bases. It also gives an accounting of all untraced requirements. Logicon is continuing to define

additional extensions to those already developed. These additional extensions would also improve the utility of PSL/PSA in the USACSC environment. Extensions to produce data-flow graphs and structure charts from the PSL data base would support USACSC's Technical Bulletin 18-103 "Software Design and Development". Other outputs can be produced which will allow for the production of documentation in accordance with DoD Standard 7939, 1-S.

3.5   PSL/PSA Generated Documentation for Communicating the Development of
       System Specifications.

This section addresses selected outputs of PSL/PSA and how they are used to
support the communication of information relating to the specification of
requirements for a target system.

3.5.1 Structure Report

The Structure Report (Volume II, Figure 3) is one of the most commonly used
reports during the requirements definition and validation phase. The report
is used to present the implied Top-down structured hierarchy of any given
element of the target system which is defined in the PSL/PSA data base with
the Part/Subpart relationship. The entire system, as defined in the data
base, can be displayed as a hierarchical decomposition of processes (system
functions) or any subset of the systems processes can be displayed by
starting at a lower level within the hierarchy. System interfaces, inputs,
outputs, or processors (a module or piece of equipment which performs a
function) can also be displayed in this manner. The report is used to
communicate the target system as a succinct aggregate of functions, items, or
information. The Structure Report may also be used throughout the develop-
ment of a system's requirements to provide the developer, the proponent
organization, and the ultimate user with a comprehensive view of the
development work.

3.5.2 Data Base Summary Report

The Data Base Summary Report provides statistical information regarding the
usage of different name types (e.g., processes, elements, etc.) and can be
used as an aid in estimating the size of the language description in a
PSL/PSA data base. The Data Base Status Report (Volume II, Figure 2) is an
extension of the basic summary report. Both reports allow the developer and
management to monitor the status of the PSL/PSA data base under development.
The Data Base Status Report also supports completeness and consistency

analysis of the target system development. In so doing it prints all of the requirements in the data base in the order in which they appear in the parent documents. It also provides pointers to the location of each requirement in the Structure Report for cross reference. This report verifies the complete coverage of the parent documents by showing the work accomplished and flagging the status of each paragraph and individual requirement contained in the data base, e.g., complete, incomplete, not found in data base, ambiguous, etc. Both the summary and the status reports may be used to succinctly communicate the current status and progress being made. They also highlight problem areas which show a lack of progress over time.

3.5.3 Process Chain Report

The process Chain Report (Volume II, Figure 4) presents, in a graphical format, the sequence of events and processes which are triggered by stimulating the system. The report is used to display the sequence of functions which make up the control or functional flow of the target system. This report may primarily be used to support the developer in performing control flow analysis and is of small consequence to the proponent or ultimate user since it is mainly representative of the internal characteristics of the system. The Process Chain Report may be used to verify that the data-flow-diagrams, approved during walk-throughs, are properly represented in the data base and the proposed design.

3.5.4 Extended Picture Report

The Extended Picture Report (Volume II, Figure 7) is used to present the flow of information into, within, and out of the target system. It presents, in graphical format, a network of data names, for each data name entered into the analyzer, which relate to the input data name by either the structure of the data or the data flow itself. For each input name one of four picture reports may be obtained. These are: structure downward, which is a top-down look at the data structure; structure upward, which is a bottom-up look at

27

the data; date-flow forward, showing the flow of data through the system; and data-flow backwards, showing the derivation of the output by tracing backwards through the system. The graphic display of system outputs, their derivation, and relationship to system inputs is very useful in supporting system walk-throughs with the user/proponent.

3.5.5 Content Report

The Content Report (Volume II, figure 6) allows the user to view entire data structures (all levels) as they are described in the PSL/PSA data base. It presents the breakdown of all lower levels of data structures for sets, inputs, outputs, entities, and groups. This report's indented list format provides a clear display of selected levels (branches) of the input/output hierarchical structure. Like the Extended Picture Report, the Content Report is very helpful in communicating the propsed inputs/outputs of the target system. Its format is very easy to understand by non-ADP personnel.

3.5.6 Data Process Report

The Data Process Report (Volume II, figure 5) displays the interaction between information (sets, input, output, entities, elements and groups) and the processes (functions) defined for the target system. It also shows the data dependencies among processes as implied by the language descriptions of these processes. This report graphically displays the interaction between information and processes in a matrix format and is very helpful in communicating these very complex and intricate relationships.

3.5.7 Other Reports

There are numerous other reports produced by the Analyzer; however this section only high-lighted those most useful in communicating the requirements definition process to persons outside and inside the development organization. Volume II of this report provides additional information on the report generating capabilities of PSL/PSA.

DAT FILM